



第二章 数据类型和表达式

第二章教学目标

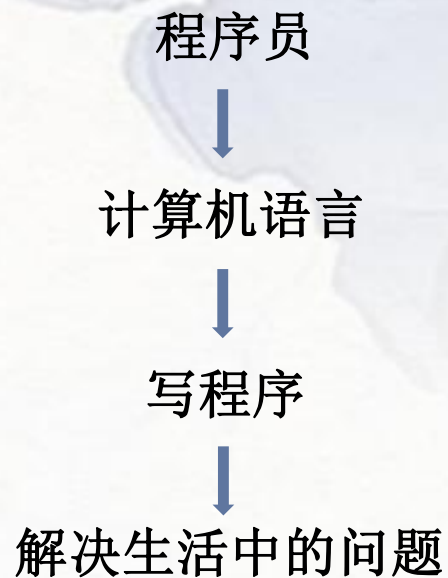
教学目标:

- 掌握C语言的基本数据类型;
- 掌握常量与变量的含义及变量的定义;
- 掌握C语言常用的表达式及运算符;
- 掌握不同数据类型间的相互转换。

教学内容:

- 2.1 C语言的数据类型
- 2.2 常量与变量
- 2.3 C语言表达式与运算符
- 2.4 数据类型转换
- 2.5 位运算

2.1 C语言的数据类型



- 描述一个商品的价格——水（2.5元）？
- 描述一个人得年级（30岁）？

基本类型

字符型 (char)

整型

基本整型 (int)

短整型 (short int)

长整型 (long int)

实型

单精度型 (float)

双精度型 (double)

2.1.1 整型数据

【例题2.1】 分析程序运行结果，理解a、b、c的含义

```
#include "stdio.h"
int main()
{
    int a,b,c;
    a=10;
    b=012;
    c=0xa;
    printf("a=%d,b=%d,c=%d\n",a,b,c);
    printf("a=%o,b=%o,c=%o\n",a,b,c);
    printf("a=%x,b=%x,c=%x\n",a,b,c);
    return 0;
}
```

"C:\Users\Administrator\Desktop\C例题\Debug\例题21.exe"

a=10, b=10, c=10

a=12, b=12, c=12

a=a, b=a, c=a

Press any key to continue

1.整型数据的表示方式

整型数据的表示

十进制数 如：10

二进制数： 存放编码：0000 0000 0000 1010

八进制数： 以0开头 如：012代表十进制数10

十六进制数： 以0x开头 如：a表示十进制数10

%d：表示以十进制整数形式输出

%o：表示以八地进制整数形式输出

%x：表示以十六进制整数形式输出

0或0x只是标识数的类型，
不输出但在内存中存放编
码相同

2 基本数据类型大小

- 每种类型的大小是多少？

sizeof() 的单位是什么？ --字节byte

```
#include <stdio.h>
```

```
int main() {
```

```
    printf("%d\n", sizeof(char));
```

```
    printf("%d\n", sizeof(short));
```

```
    printf("%d\n", sizeof(int));
```

```
    printf("%d\n", sizeof(long));
```

```
    printf("%d\n", sizeof(long long));
```

```
    printf("%d\n", sizeof(float));
```

```
    printf("%d\n", sizeof(double));
```

```
    return 0;
```

```
}
```



计算机中的单位

bit 一比特位

byte—字节=8bit

kb—1024byte (2^{10})

mb—1024kb

gb—1024mb

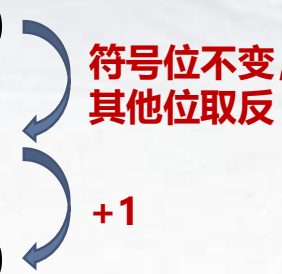
tb—1024gb

pb—1024tb

3.整型数据在内存中的存放形式

数在计算机中以二进制数的补码存放且区分为有符号数和无符号数。例如：-8和8两个数的补码存放形式如下。

-8原码：10000000 00000000 00000000 00001000
-8反码：11111111 11111111 11111111 11110111
-8补码：11111111 11111111 11111111 11111000



符号位不变,
其他位取反

+1

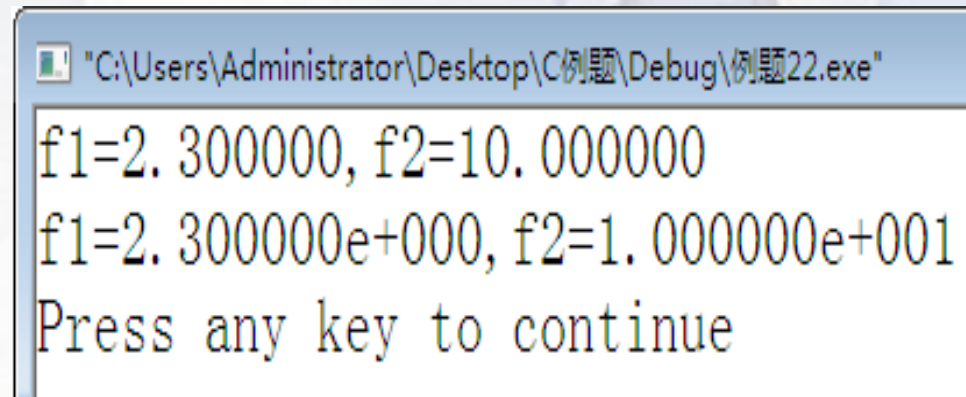
8补码：00000000 00000000 00000000 00001000

总结：正整数的原码、反码、补码相同。

2.1.2 实型数据

【例题2.2】分析程序，体会符号f1、f2的含义

```
#include "stdio.h"
int main()
{
    float f1;
    double f2;
    f1=2.3;
    f2=10.0;
    printf("f1=%f,f2=%f\n",f1,f2);
    printf("f1=%e,f2=%e\n",f1,f2);
    return 0;
}
```



```
"C:\Users\Administrator\Desktop\C例题\Debug\例题22.exe"
f1=2.300000, f2=10.000000
f1=2.300000e+000, f2=1.000000e+001
Press any key to continue
```


1.实型数据的表示法

实型数据的表示

十进制小数：由数字和小数点组成。2.3，10.0

指数形式：以规范化指数形式表示。

表示方法：小数点必须一个非0的数字，小数点后以E表示的指数形式。

例如：0.23的规范化指数形式为：2.3E-001

10.0的规范化指数形式为：1.0E+001

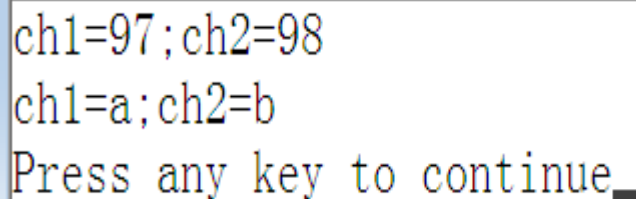
%f：以十进制小数格式输出带6位小数位

%e：以规范化的指数形式输出

2.1.3 字符型数据

【例题2.3】分析下列程序，体会ch1、ch2的含义

```
#include "stdio.h"
int main()
{
    char ch1, ch2;
    ch1 = 'a';
    ch2 = 'b';
    printf("ch1=%d;ch2=%d\n", ch1, ch2);
    printf("ch1=%c;ch2=%c\n", ch1, ch2);
    return 0;
}
```



```
ch1=97;ch2=98
ch1=a;ch2=b
Press any key to continue.
```

%c: 表示以字符格式输出

97和a在内存的编码相同，a的ascii是97

1.字符

字符型数据是一对单撇括起来的单个字符，如：'a'、'D'、'2'等称为字符型数据。对于字符型数据，一般不考虑它的符号问题。

在C语言中，字符型数据用char来表示，每个字符数据以字符的ASCII码在计算机内存中占用8个二进制位（1字节）。

字符 'a' 的存放形式

0	1	1	0	0	0	0	1
---	---	---	---	---	---	---	---

97

2.转义字符

```
2  #include <stdio.h>
3  int main(){
4      printf("c:\test\test\n");
5      printf("%s\n", "abc");
6      printf("c:\\test\\test\n");
7      printf("%c\n", '\\');
8      printf("%c\n", '\"');
9      printf("%c\n", '\0');
10
11 }
```

C:\Windows\system32\cmd.exe

c: est est

abc

c:\test\test

来源(S): 生

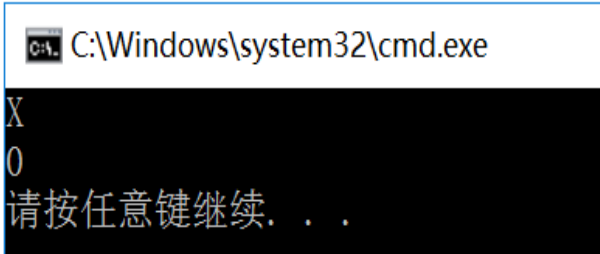
请按任意键继续. . .

这里的 `'\t'` 变成了转义字符，
转变成了其他意思。

转义字符	意义
<code>\n</code>	回车换行符号
<code>\t</code>	水平制表（跳到下一个 Tab 位置）
<code>\b</code>	退格，将当前位置移到前一个
<code>\r</code>	回车，将当前位置移到本行开头
<code>\f</code>	换页，将当前位置移到下页开头
<code>\a</code>	响铃符号
<code>\\</code>	反斜杠（\）
<code>\'</code>	单引号字符
<code>\"</code>	双引号字符
<code>\0</code>	空字符
<code>\ddd</code>	1~3 位 8 进制数所代表的字符
<code>\xhh</code>	1~2 位 16 进制数所代表的字符

2.转义字符

```
10 printf("%c\n", '\\130');
11 printf("%c\n", '\\x30');
12
13 }
```



- 八进制130 -- 十进制88 --X的ascall码是88
- 十六进制30 -- 十进制48 --字符0的ascall码是48

ASCII打印字符										
0011		0100		0101		0110		0111		
3		4		5		6		7		
十进制	字符	十进制	字符	十进制	字符	十进制	字符	十进制	字符	Ctrl
48	0	64	@	80	P	96	`	112	p	
49	1	65	A	81	Q	97	a	113	q	
50	2	66	B	82	R	98	b	114	r	
51	3	67	C	83	S	99	c	115	s	
52	4	68	D	84	T	100	d	116	t	
53	5	69	E	85	U	101	e	117	u	
54	6	70	F	86	V	102	f	118	v	
55	7	71	G	87	W	103	g	119	w	
56	8	72	H	88	X	104	h	120	x	
57	9	73	I	89	Y	105	i	121	y	
58	:	74	J	90	Z	106	j	122	z	
59	;	75	K	91	[107	k	123	{	
60	<	76	L	92	\	108	l	124		
61	=	77	M	93]	109	m	125	}	
62	>	78	N	94	^	110	n	126	~	
63	?	79	O	95	_	111	o	127	△	*Backspace 代码：DEL

3.字符串

- 在C语言字符串是以双引括起来的单个字符或多个字符。
- 如: "How are you", "a"称为字符串。
- 字符串的长度就是字符的个数。
- 长度为n的字符串, 在计算机的内存存储时占n+1个字节, 分别存放各字符的编码, 最后一个字节是NULL字符 (称空字符, 该字符在ASCII字符集中的编码是0, 为了书写方便, 在C语言中用'\0'来表示) 。

'a' 字符的存放

0	1	1	0	0	0	0	1
---	---	---	---	---	---	---	---

"a" 字符串的存放

0	1	1	0	0	0	0	1	\0
---	---	---	---	---	---	---	---	----

2.2 常量与变量

C语言程序中的数据，不论是何类型，均是以常量或变量的形式表现。

2.2.1 常量

常量是指在程序运行过程中其值不能改变的量。

直接常量：程序中不作任何说明直接使用的常量。

符号常量：程序的开头用#define定义后才能使用量的常量。

1.直接常量

直接常量按其数据类型分整型常量、实型常量、字符常量和字符串常量。

常量类型

整型常量 如: 100; 0123; 0x123

实型常量 如: 123.456; 1.23456E+002

字符常量 如: 'a'; 'A'; '\n'

字符串常量 如: "computer"

- 字符常量是用单引括起来的单个字符;
- 字符串常量是用双撇括起来的多个字符序列, 在计算机内存存储时以 '\0' (ASCII) 作为字符串结束标志。

//const 修饰的常变量

const float pai = 3.14f; //这里的pai是const修饰的常变量

pai = 5.14; //是不能直接修改的!

2.符号常量

符号常量是使用宏命令#define定义的标识符。

#define 标识符 常量表达式

【例题2.5】

```
#define PI 3.14159
#include <stdio.h>
int main()
{
    float r,s;
    printf("请输入半径: ");
    scanf("%f",&r);
    s=PI*r*r;
    printf("圆的面积为: %f\n",s);
    return 0;
}
```

2.2.2 变量

变量是指在程序运行过程中其值是**可以改变**的量，变量有三种属性：变量名、变量类型和变量值。

```
int age = 150;  
float weight = 45.5f;  
char ch = 'w';
```

- 遵循标识符的命名规则
- C语言保留字不能作变量名
- 区分大小写

【例题2.6】编写程序：在屏幕上输出如下结果。

a=3,b=5,c=8

x=14.52,y=125.786,40.1

```
#include "stdio.h"
int main()
{
    int a=3,b,c;
    float x,y,z;
    b=5;
    c=8;
    printf("Input x,y,z:\n");
    scanf("%f%f%f",&x,&y,&z);
    printf("a=%d;b=%d;c=%d\n",a,b,c);
    printf("x=%.2f;y=%.3f;z=%.1f\n",x,y,z);
    return 0;
}
```

2.3 表达式和运算符

描述不同运算的符号称为运算符。由运算符把操作数连接起来的式子称为表达式。C语言特点之一是运算非常丰富，主要有以下运算符。

C语言运算符

算术运算符： +、-、*、/、%

赋值运算符： =、+=、-=、*=、/=

自增自减运算符： ++ --

逗号运算符： ,

条件运算符： ?

2.3.1 算术运算符与算术表达式

常用算术运算符

对象数目	运算符	含义	举例	结果
双目	+	加	$a+b$	a 与 b 的和
	-	减	$a-b$	a 与 b 的差
	*	乘	$a*b$	a 与 b 的乘积
	/	除	a/b	a 除 b 的商
	%	求余	$a\%b$	a 除 b 的余数
单目	+	正号	$+a$	a 的值
	-	负号	$-b$	a 的算术负值

2.3.2 赋值运算符与赋值表达式

1. 赋值运算符

"="

2. 赋值表达式

变量=表达式

a=2;

3. 复合赋值运算符

+=、-=、*=、/=、%=、<<=、>>=、&=

例如:

a+=10

x/=y-5

m*=n+5

等价于

a=a+10

x=x/(y-5)

m=m*(n+5)

2.3.2 赋值运算符与赋值表达式

分析下列复合赋值表达式的求解过程

$a += a -= a * a$ 其中 $a = 10$

规则是从右起的第一个复合赋值符号开始，依次致左进行

① $a -= a * a$ 的运算为 $a = a - (a * a)$, 则 $a = -90$

② $a += a$ 的运算为 $a = a + a$, $a = -180$

2.3.3 自增自减运算符

自增 (++)、自减 (--) 运算符的作用是使变量自增1和自减1

- ++i、--i 先使i的值加（减）1，再参与其他运算
- i++、i-- 先让i参与其他运算，再使i的值加（减）1

【例题2.7】

```
#include<stdio.h>
int main()
{
    int i,j,m,n;
    i=5;
    j=15;
    m=i++;
    n=++j;
    printf("i=%d, j=%d, m=%d, n=%d\n",i,j,m,n);
    return 0;
}
```

程序执行后的结果：

i=6,j=16,m=5,n=16

2.3.4 逗号运算符

表达式1, 表达式2, ..., 表达式n

- 求解过程：依次求解表达式1, 表达式2, ..., 表达式n的值
- 整个逗号表达式的值为表达式n的值
- 逗号运算符的优先级是所有运算符中最低

例如：y=(x=2,5*6) 则y=30

2.3.5 条件运算符及条件表达式

表达式1 ? 表达式2 : 表达式3

- 求解过程：先求解表达式1，若表达式1的值为真，结果为表达式2的值
若表达式1的值为假，结果为表达式3的值。

例如：max= (a>b) ? a : b (a=4,b=5)

max=5

2.4 数据类型转换

2.4.1 自动转换

由编译系统自动完成的数据类型的转换称自动转换。

例子： $8+x*y+ 'a'$ (x为int型变量, y为float型变量)

1. 先算 $x*y$, 将x、y都转化为double型, 运算结果为double型;
2. 再8与 $x*y$ 的和, 将8转换为double型, 结果为double型;
3. 最后加上字符a, 先将a转换成整数97, 再转换为double型, 结果为double型。

2.4.2 强制转换

强制类型转换符是由类型名加一对圆括号构成，功能是强制将一个表达式结果的类型转换为指定类型。其一般形式：

(类型名) 表达式

【例题2.11】 `#include <stdio.h>`
`int main()`
`{`
 `int a=5,b;`
 `float c;`
 `b=a/2;`
 `c=(float)a/2;`
 `printf("b=%d,c=%f\n",b,c);`
 `return 0;`
`}`

程序执行结果为：
b=2,c=2.500000

本章小结

- 掌握C 语言数据类型中的整型、实型和字符型；
- 掌握整型常量、实型常量、字符常量和字符串常量；
- 掌握变量的定义方法；
- 掌握C 语言的算术运算符、赋值运算符、自增自减运算符、逗号运算符和条件运算符的应用。